Remarks/Arguments

The Examiner is thanked for the careful review of this Application. Claims 1-22 are pending after entry of the present Amendment. Amendments were made to claims to correct typographical errors and in response to the Office's new grounds of rejection. The amendments do not introduce new matter.

Rejections under 35 U.S.C. § 103(a):

Claims 1-8 and 10-13 have been rejected under 35 U.S.C. section 103(a) as being obvious over the U.S. Patent No. 6,711,616 to Stamm et al. (Stamm) in view of the U.S. Patent No. 5,371,883 to Gross et al. (Gross), and further in view of U.S. Patent No. 6,009,454 to Dummermuth; Claim 9 has been rejected over Stamm in view of Gross, in view of Dummermuth, and further in view of U.S. Patent No. 6,112,243 to Downs et al. (Downs); Claims 14 and 15 have been rejected over Stamm in view of Dummermuth; Claims 16-17 have been rejected over Stamm in view of Dummermuth, further in view of Gross; Claims 18-20 have been rejected over Stamm in view of the U.S. Patent No. 4,800,521 to Carter et al. (Carter), further in view of Dummermuth; Claim 21 has been rejected over Stamm in view of Carter further in view of Dummermuth, and further in view of Downs; and Claim 22 has been rejected over Stamm in view of Carter in view of Dummermuth, and further in view of Gross. For at least the reasons provided below, none of the combinations of the cited prior art raises a *prima facie* case of obviousness against the subject matter defined in amended independent claims 1, 5, 14, and 18.

The Office has explicitly admitted that Stamm, Gross, and Carter (all of which were previously cited by the Office in the prior Office Action) fail to disclose, teach, or suggest allocating computer resources for tasks prior to proceeding to the next operation. In maintaining the Office's rejection under 35 U.S.C. section 35(a), the Office has cited to Dummermuth interpreting that Dummermuth discloses the allocating computing resources to tasks prior to proceeding to the next operation feature of the claimed invention. It is respectfully submitted that the Office's reliance on Dummermuth is misplaced as Dummermuth fails to disclose <u>allocating of processing resources for each task prior to proceeding to the next operation</u> feature of the claimed invention.

Specifically, Dummermuth discloses a multi-tasking operating system that includes a single operating system, a <u>single processor</u>, and a memory. In achieving its goal, Dummermuth focuses on using the <u>single processor</u> to process the tasks on the basis of the <u>number of instructions</u>, and not on the basis of time. In doing so, Dummermuth discloses executing a <u>predetermined</u> <u>number of instructions of each task</u> by the processor at any given time before the processor switches <u>from the task being executed to another task</u>.

In rejecting the claimed invention, as defined in independent claims 1, 5, 14, and 18, the Office has cited to column 8, lines 44-49 of Dummermuth for support, which provides:

> The ability to <u>select the number of instructions executed in each task</u> before the <u>next task is switched to</u>, allows precise allocation of processor resources to particular tasks, keeping with the strict demands fore processor efficiency imposed by an industrial control system which operates in real-time. [Emphasis added.]

In contrast to the Office's interpretation, the latter excerpt discloses, suggests, or teaches executing only a <u>predetermined number of instructions at a time for each task</u>, and not until each aspect of the task is executed or until an aspect of the task causes the execution of the task to discontinue. In fact, in Dummermuth, once the predetermined number of instructions for a given task has been executed, Dummermuth switches to the next task. That is, the execution of any given task is <u>stopped multiple times, and the processor</u> switches to the execution of the next task while the given task still has more instructions that have to be executed.

Additionally, Dummermuth discloses, teaches, or suggests <u>sharing a single processor</u> by multiple tasks. In fact, Dummermuth uses the functionalities available to a single processor to keep track of the instructions that have been executed and the instructions that have to be executed next. In contrast to the Office's interpretation, nowhere does Dummermuth disclose, teach, or suggest using more than one processor or managing the tasks and instructions during the partial execution of each task using a first computing resource and a second computing resource (i.e., more than one processor). In the claimed invention, however, tasks can be executed by a plurality of processing resources. A main task is executed by a first processing system and the subtask is executed by a second processing system. Additionally, as defined in independent claims 5 and 14, the first processing resource is different than the second processing resource.

Considering Durmmermuth's failure to disclose, teach, or suggest the latter features of the claimed invention as well as the above-referenced admission by the Office, it is respectfully submitted that each and every combination of the cited prior art still fail to disclose, teach, or suggest all the features of the claimed invention. For instance, among other features, the combination of the cited prior art fails to disclose, teach, or suggest, that the allocated computing <u>resources includes a first computing system and a second computing system</u>, the <u>first processing resource is separate from the second processing resource</u> (as defined in claim 5), or <u>allocating a separate processing resource to execute each task of the plurality of tasks</u> (as defined in independent claim 14). The combination of the cited prior art also fails to disclose that <u>the first computing resource is configured to be allocated to the main task until each aspect of the main task is executed or an aspect of the main task causes</u>

the execution of the main task to discontinue and that the second computing resource is configured to be allocated to the subtask until each aspect of the subtask is executed or an aspect of the subtask causes the execution of the subtask to discontinue.

Additionally, each combination of the cited prior art further fails to disclose, teach, or suggest a code of main task having program instructions for requesting loading of a code for a subtask to a second computing system, the code for the main task having client-server communication with the code with the code for the main task such that the code for the main task receives processing results directly from the code for the subtask, deploying of a subtask to the second processing resource once a special request for the subtask is received from the main task, enabling communication between the first task and the subtask, providing the main task with test results of the subtask, dispatching the special request to the system controller, or searching the task list for the subtask having the requested attributes.

Furthermore, if Stamm, Gross, and Dummermuth were combinable (a proposition with which the Applicants disagree), the resulting system and method still fails to disclose, teach, or suggest the claimed invention, as defined in independent claims 1 and 5. For instance, in Gross, the control program forwards instructions from a test case to test machines for execution. Additionally, in Stamm, the control program maintains control of the sequence of execution. Thus, if Stamm and Gross were to be combined, the control program would be maintaining the control of the sequencing of the tasks and subtasks to be executed, and not the main task.

Furthermore, based on the teachings of Dummermuth, only a predetermined number of instructions for each task would be executed, and not all the aspects of each task. Still further, even if a predetermined number of instructions of each task were to be executed by a respective processing resource, nothing in Stamm, Gross, and Dummermuth discloses, teaches, or suggests the manner the execution of each task will and can continue from the instruction following the last executed instruction in each task. Still further, in Dummermuth, only a single processor is used. As such, Dummermuth fails to provide any disclosure, teaching, or suggestion the manner execution of different tasks on different computing resources can be managed, stopped after the execution of a few instructions, and continue execution from the correct instruction for each task in each execution cycle.

Additionally, Stamm, Gross, Dummermuth, and Downs fail to disclose, teach, or suggest that a main task can send a special request for a subtask, as defined in the claimed invention. As such, even if a main task in Stamm or Gross could send a request for a specific subtask (a proposition with which the Applicants disagree), the control program overrides the special request by the main task as the control programs in both, Stamm and Gross control

the sequencing of the tasks and subtasks. Furthermore, even then, only a predetermined number of instructions are executed, and not all aspects of each task.
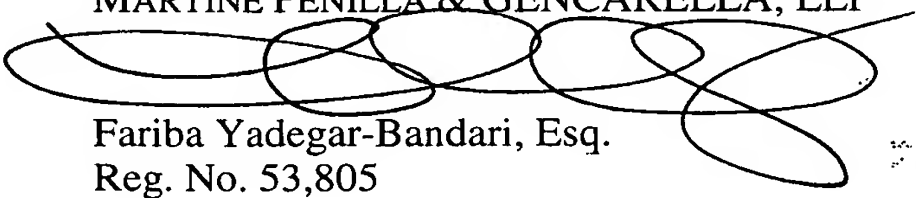
Still further, if Stamm were to be modified using the teachings of Carter and Dummermuth, the resulting system and method still fails to disclose, teach, or suggest the claimed invention, as defined in independent claim 18. For instance, the combination still fails to disclose, teach, or suggest deploying a second task to a second processing resource upon receiving the result of the execution of the first task. Rather, Carter only discloses a multiprocessor that has two portions in communication with one another or that each processing resource is allocated to a respective task until each aspect of the task is executed or an aspect of the task causes the task to discontinue. Furthermore, even if the Office's interpretation with respect to the combination of Stamm, Carter, and Dummermuth were true, only a predetermined number of instructions for each task would be executed in the resulting system. In fact, only the result (if any) of execution of a predetermined number of instructions could be sent to the main task, and not the result of execution of the first task.

Additionally, nothing in Downs can cure such deficiencies in Stamm, Carter, Gross, or Dummermuth. Therefore, it is respectfully submitted that claims 1-22 are patentable under 35 U.S.C. section 103(a) over any combination of the cited prior art.

The Applicants hereby submit that the subject Amendment complies with 37 C.F.R. 1.116(c) as the Office's new grounds of rejection necessitated the amendments to the claims. As such, it is respectfully requested that amendments to the claims be entered.

The Applicants respectfully submit that all of the pending claims are in condition for allowance. Accordingly, a notice of allowance is respectfully requested. If the Examiner has any questions concerning the present Amendment, the Examiner is kindly requested to contact the undersigned at (408) 774-6913. If any additional fees are due in connection with filing this Amendment, the Commissioner is also authorized to charge Deposit Account No. 50-0805 (Order No. SUNMP031). A duplicate copy of the transmittal is enclosed for this purpose.

Respectfully submitted,
MARTINE PENILLA & GENCARELLA, LLP

Fariba Yadegar-Bandari, Esq.
Reg. No. 53,805

710 Lakeway Drive, Suite 200
Sunnyvale, CA 94085
Telephone (408) 774-6913
Facsimile (408) 749-6901
Customer No. 32291